Towards Multi-Functional ECG Smart System Based on a Client-Edge-Cloud Architecture

Rajdeep Kumar Nath
VTT Kuopio, Finland
rajdeep.nath@vtt.fi

Jaakko Tervonen
VTT Espoo, Finland
jaakko.tervonen@vtt.fi

Johanna Närväinen
VTT Kuopio, Finland
johanna.narvainen@vtt.fi

Kati Pettersson
VTT Espoo, Finland
kati.pettersson@vtt.fi

Jani Mäntyjärvi
VTT Oulu, Finland
jani.mantyjarvi@vtt.fi

***Abstract*—This paper presents a novel client-edge-cloud-based framework that integrates the learning of task-invariant ECG feature representations from ultra-short ECG segments (<10 sec) and subsequent training of task-specific machine learning (ML) classifiers for different applications. Our proposed framework removes the need for application-specific ECG processing by training a general ECG representation learner in a self-supervised manner. The ECG representation learner is then used for generating feature inputs for the different task-specific applications. The proposed framework distributes the computation across cloud, edge, and client components depending on the resource requirement and time criticality. We demonstrate the feasibility and promise of the proposed approach on two different applications, that is, acute stress type classification, and biometric user identification and authentication. The use cases were analyzed using the computational parameters for the different models and computational tasks along with the overall performance. Our analyses show that the application-specific ML models can perform real-time inference in less than a second and the training time of the ML classifiers at the edge devices are in the order of 10-20 seconds. In the future, the proposed framework can be utilized for developing reliable, secure, and multi-functional ECG-based smart systems.**

***Index Terms*—Acute stress, Electrocardiogram (ECG), Contrastive Learning, Smart Healthcare, Biometrics, convolutional neural network (CNN), Client-Edge-Cloud, Wearable Computing**

I. INTRODUCTION

Electrocardiogram (ECG) is a reliable and valuable indicator of cardiovascular health and it is used as a reference test for various clinical diagnoses. Rapid development in the area of sensor electronics has made it possible to record ECG signals continuously and unobtrusively in a real-world setting. As a result, adapting artificial intelligence (AI) techniques in the automated processing and analyses of ECG signals for smart health diagnosis has been an active area of research in recent years. AI-based analyses of ECG signals have found applications in diagnosing and monitoring several cardiovascular abnormalities such as detecting paroxysmal atrial fibrillation (PAF) [1] and mental stress and anxiety [2]. Apart from applications in health diagnosis, ECG has also demonstrated promise for use in biometric systems [3], [4]. Hence, the ECG signal is an important biosignal and ECG-based smart systems can be used for a variety of applications to improve overall health and enhance the quality of life.

Despite the huge potential of ECG signals, there are several factors and practical challenges that can limit the performance and practicality of ECG-based smart systems in a real-world setting. Some of these are the acquisition of good-quality ECG signals and the availability of reliable labeled data for creating the health diagnostic model. Furthermore, ECG signal patterns are affected by various factors such as existing health conditions and response to psychological and physiological stimuli [5], [6]. Depending on the application, usually, different types of ECG features are used to capture the application-specific dynamics for developing accurate diagnosis and inference. For example, heart rate variability (HRV) features are commonly used in the context of mental health applications, however, for biometric applications, only the features around the R peaks of ECG signal are mostly used [3], [4], [7]. Because of this, developing an accurate ECG-based smart system is challenging and requires domain knowledge expertise for specific applications.

Furthermore, there are challenges in the real-world implementation of ECG-based smart systems intended for multiple applications. ECG signals are usually sampled at higher sampling rates ($>$ 100 Hz) and can consume a lot of bandwidth when transferred from the sensor device to external resources for processing. In addition, the implementation of multiple applications in an ECG-based smart system would require application-specific processing of the ECG signal which can increase the need for higher computational resources at the edge. In this context, client-edge-cloud-based architecture for healthcare applications has been gaining popularity in recent years [8]. A client-edge-cloud architecture is suitable in these kinds of applications since the computation and data movement can be distributed to the different components such as client, edge, and cloud according to the resource needs and criticality of the application. For example, the inference model for an ECG-based biometric authentication system should be placed at the client's side such as smartphones. Whereas the inference model for less time-critical applications can be deployed on the edge device which can periodically communicate with the client side.

In this paper, we present a novel framework based on client-edge-cloud architecture for developing ECG-based smart systems intended for multiple applications in a real-world setting

The work was funded by the Academy of Finland under GrantNos: 334092,313401,351282, RM4Health resorting under ITEA, funded by Business Finland, and VTT

(Figure 1). The proposed framework uses self-supervised representation learning to create a model for generating the feature representation of ultra-short ECG signal segments. Since the ECG representations are learned without associated labels in a self-supervised manner, it has the potential of generalizing ECG representations for multiple applications thereby eliminating the need for using application-specific feature engineering approaches. The cloud resource of the framework hosts a large database of unlabeled ECG signals that are used to train an ECG representation model. The trained ECG representation model is then imported to an edge device where different prediction models are trained based on the features generated by the ECG representation model. Finally, on the client side, real-time ECG signal segments from ECG sensor nodes are forwarded to the edge device through a client device such as a smartphone or other health monitoring systems. The main contribution of our work is as follows:

- We propose a client-edge-cloud-based framework for learning task-invariant ECG signal representation and subsequently perform task-specific classifications.
- We analyze the feasibility of the proposed framework in terms of computational parameters and overall performance on two unrelated use cases.

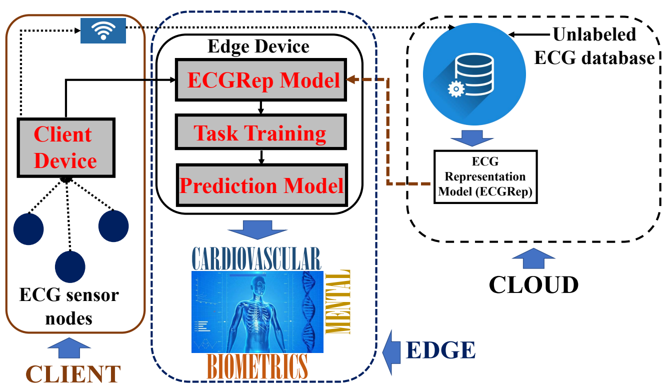

Fig. 1. Visualization of the proposed framework for real-world implementation of ECG-based smart system for several downstream applications in a client-cloud-architecture setting.

II. Proposed Work

The overview of the proposed framework is shown in Figure 2. The three components, client, edge, and cloud are shown as three different blocks from left to right. Details on the technical details and functionality of the three components are discussed below:

A. Cloud Component

The main functionality of the cloud component is to train the CNN encoder using a simple framework for contrastive learning (SimCLR) [9]. The CNN encoder is trained to generate 128-length feature vectors from ultra-short ECG segments (<10 sec). Details on the training scheme and architecture of the CNN encoder are presented in our recent research work [10]. The feature vectors are the higher-level representations learned from the ECG segment. Figure 3 shows an example of a raw ECG signal segment of 8-second length (top) and its corresponding feature vector (bottom) of length 128. Subsequently, the trained CNN encoder is then packaged for exporting to edge/client modules.

B. Edge Component

The edge component consists of the trained CNN encoder (imported from the cloud) and a subroutine for training machine learning (ML) classifiers for specific tasks. The trained CNN encoder takes ECG segments as input and outputs the feature representation which is used for training the ML classifiers (Figure 2). The detection models are then configured to run in detection mode at the edge device itself and or exported to a client device such as a user smartphone or other personal health monitoring devices depending on the computing resource of the client device. Since edge devices are resource constraints, very few labeled training examples are used for training the ML classifiers.

C. Client Component

The client component consists of wearable sensor devices for ECG signal acquisition and client devices for real-time inference and notifications. The sensor devices aggregate short ECG segments and perform basic digital signal processing such as analog-to-digital conversion and filtering of the raw signals. Subsequently, the short raw ECG segments are fed as input to the CNN encoder for representation generation and inference using the detection models either on a client device such as a smartphone or tablet or to an edge device. Since the client and edge components are very near to each other (bi-directional red arrow in Figure 2), the inference tasks can be distributed to either of these components depending on the requirement, and processing capability of client devices, and the specific application.

The training of the CNN encoder for learning the ECG feature representation is the most computationally heavy task and hence is performed using cloud resources. The specific computational parameters associated with ECG representation learning such as model size, number of weighted parameters, and training time will depend on several factors such as CNN encoder model architecture, and the size of the dataset used for training. However, these parameters are still by far expected to be higher than that of ML classifiers used for developing the detection models and hence are suitable for cloud computation. To analyze the practicality of the proposed framework, we validated our work on two unrelated use cases: (i) Classification of acute stress types, and (ii) Biometric system (user identification and authentication)

D. Usecase analyses

The study design and dataset used for the use case analyses are discussed in our previous work [11]. The objective of the classification of the first use case was to develop a detection model that can distinguish between rapidly alternating physical and mental stress tasks [12]. For the second use case, the

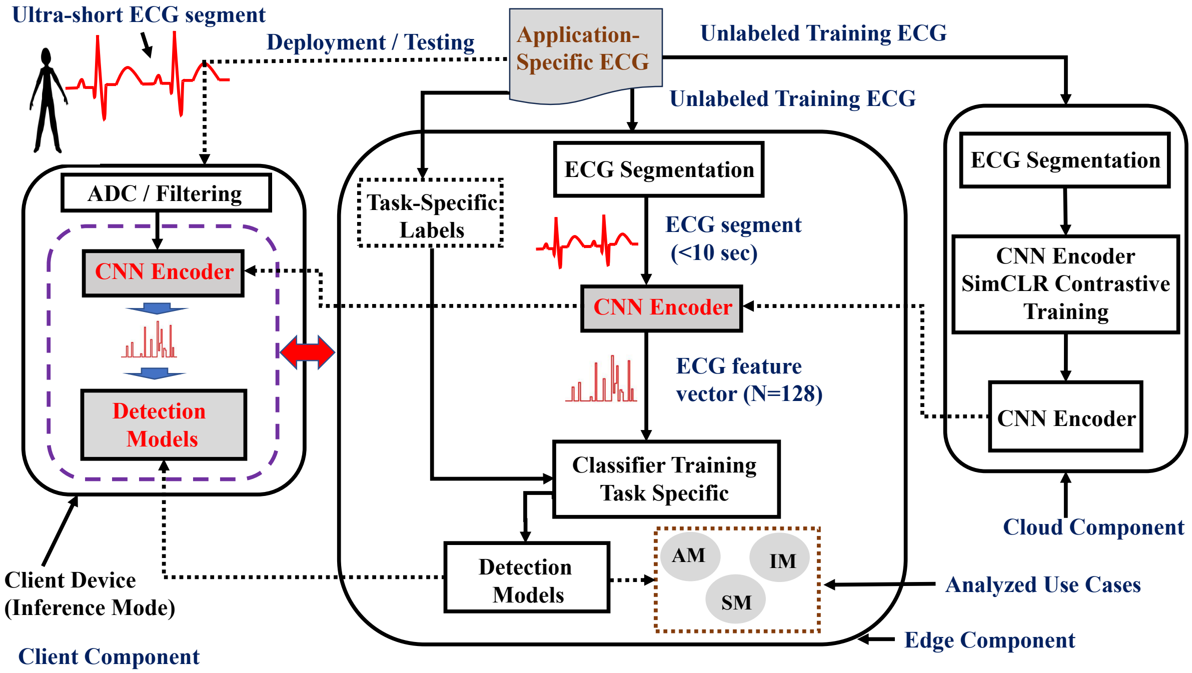

Fig. 2. Overview of the proposed work. A CNN encoder is trained using cloud resources for generating ECG representations. The trained CNN encoder is imported to an edge environment where task-specific machine learning classifiers are trained and subsequently deployed in a client ecosystem in inference mode. The brown dotted box inside the edge components shows the three ML classifiers implemented for our use case analyses. AM = Authentication Model, IM = Identification Model, SM = Stress Model

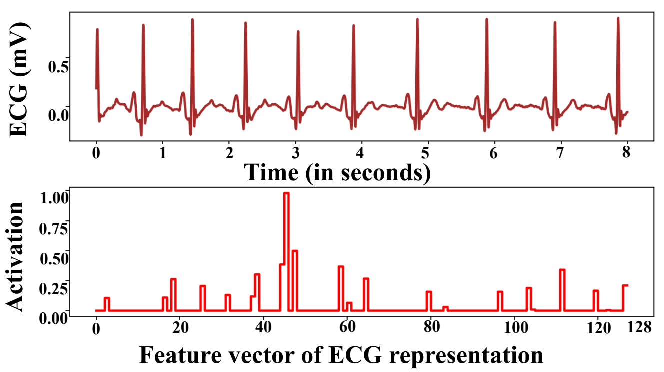

Fig. 3. An example of raw ECG signal segment of 8 second length (top) and its corresponding representation generated by the CNN encoder (bottom). Similar representations are obtained from 2 second length ECG segments for biometric identification and authentication use case

biometric system, there are two objectives: (i) develop a user identification model that uniquely identifies a user from other users enrolled in the biometric system, and (ii) develop a user authentication model that can authenticate the identity of a particular enrolled user (Figure 2). The simulation of the different computational parameters (Table I) are based on the processing performed on a laptop having a i5-1245U processor with a speed of 1.60 GHz and 16 GB RAM. The detection models operate on raw ECG signal segment sampled at 250Hz acquired with a single channel ECG measurement setup [10], [11]. For our use case analyses, the trained CNN encoder had a total of 411,584 weighted parameters with a size of 1.41 MB and a training time of 3 hours (Table I). As the size of the CNN encoder is low and the inference time per ECG segment is significantly lower than 1 second, the encoder can be efficiently deployed in an edge and client device for real-time inferences. All the detection models have significantly lower training time (order of 10s of seconds) and fast inference time per ECG segment ($<$ 1 sec) which are ideal parameters for real-time inference and feedback.

III. Discussions and Future Work

This paper presented a novel client-edge-cloud-based framework for developing ECG-based computing. Our proposed ECG-based solutions are intended for multiple ECG-related applications. Our framework uses a self-supervised representation learning technique to create an ECG representation learner that learns key information from ultra-short ECG signal segments ($<$10 seconds). The representation learner is capable of generating ECG feature representations that can generalize to multiple applications. We evaluate and demonstrate the effectiveness of our proposed approach on two completely different use cases, acute stress type classification, and ECG-based biometric systems. This is a work in progress and our preliminary results show the potential of the proposed method for learning general ECG signal representations that can be accurately used for simultaneous tasks.

Our proposed solution can significantly reduce the computational resources that are required for building ECG-based

TABLE I

COMPUTATIONAL PARAMETERS FOR THE DIFFERENT DETECTION MODELS AND THE CNN ENCODER. NO. PARAMS = NUMBER OF WEIGHTED PARAMETERS IN THE TRAINED MODEL. $\sim$ DENOTES VALUES THAT ARE AGGREGATED OVER SEVERAL ITERATIONS

Models	No. Params	Size (MB)	Training Time	Inference Time (s)	Use case Accuracy	ECG segment length (s)
CNN Encoder	411,584	1.41	$\sim$3 hours	0.015	NA	8/2
Acute Stress Model	26,914	1.45	$\sim$15 sec	0.115	73%	8
Authentication Model	$\sim$60000	1	$\sim$13 sec	0.08	$\sim$99%	2
Identification Model	762,980	34.3	$\sim$10 sec	0.07	98%	2

smart systems with multiple functionalities by performing most of the heavy training in the cloud. Although labeled training examples are used in the edge devices for training the detection models, the number of training examples required is very small compared to the number of training examples required for training the representation model. For example, for developing the acute stress type detection model, only 20% of labeled training data was used and still, an accuracy of 73% was obtained (Table I). The performance was significantly better when compared with the stress detection model that was developed using stress-specific ECG features [10]. A similar observation was also observed with the user identification and authentication system [13] when only about 25% of labeled training data was used to train the user identification and the authentication model. This indicates that the proposed framework is promising in situations where the availability of reliable labeled training data is questionable.

Raw ECG signals can be vulnerable to attacks during transmission and even during storage. Hence, if the ECG signal is compromised, an unauthorized person can get access to the ECG signal patterns which can reveal sensitive health information. Furthermore, if the ECG signals are associated with labels that indicate certain health conditions, there are high chances of privacy breaches if the data are on public cloud servers. Our proposed solution can address this problem by encoding the raw ECG signal in real-time before transmitting it to an external resource. The labels associated with ECG signals are stored locally within the edge server. Since, the cloud server only contains raw, unlabeled, and anonymous ECG signal segments, the consequences in case of a data breach will be less severe. In addition, since our proposed framework can encode a relatively high number of raw ECG samples into smaller 128-length feature vectors in real time, the bandwidth required for data transmission can also be significantly reduced. Our future work will focus on the full scale implementation of the proposed framework for real time implementation. Furthermore, we will investigate and integrate additional ECG applications within the proposed framework.

REFERENCES

[1] B. Pourbabaee, M. J. Roshtkhari, and K. Khorasani, "Deep convolutional neural networks and learning ecg features for screening paroxysmal atrial fibrillation patients," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 12, pp. 2095–2104, 2017.

[2] J. Henry, H. Lloyd, M. Turner, and C. Kendrick, "On the robustness of machine learning models for stress and anxiety recognition from heart activity signals," *IEEE Sensors Journal*, 2023.

[3] P. Melzi, R. Tolosana, and R. Vera-Rodriguez, "Ecg biometric recognition: Review, system proposal, and benchmark evaluation," *IEEE Access*, 2023.

[4] A. N. Uwaechia and D. A. Ramli, "A comprehensive survey on ecg signals as new biometric modality for human authentication: Recent advances and future challenges," *IEEE Access*, vol. 9, pp. 97 760–97 802, 2021.

[5] B. Pourbabaee, M. J. Roshtkhari, and K. Khorasani, "Deep convolutional neural networks and learning ecg features for screening paroxysmal atrial fibrillation patients," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 12, pp. 2095–2104, 2017.

[6] F. Agrafioti, D. Hatzinakos, and A. K. Anderson, "Ecg pattern analysis for emotion detection," *IEEE Transactions on affective computing*, vol. 3, no. 1, pp. 102–115, 2011.

[7] I. Odinaka, P.-H. Lai, A. D. Kaplan, J. A. O'Sullivan, E. J. Sirevaag, and J. W. Rohrbaugh, "Ecg biometric recognition: A comparative analysis," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 6, pp. 1812–1824, 2012.

[8] A. Hennebelle, H. Materwala, and L. Ismail, "Healthedge: A machine learning-based smart healthcare framework for prediction of type 2 diabetes in an integrated iot, edge, and cloud computing system," *Procedia Computer Science*, vol. 220, pp. 331–338, 2023.

[9] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[10] R. K. Nath, J. Tervonen, J. Närväinen, K. Pettersson, and J. Mäntyjärvi, "Towards self-supervised learning of ecg signal representation for the classification of acute stress types," in *Proceedings of the Great Lakes Symposium on VLSI 2023*, 2023, pp. 85–90.

[11] K. Pettersson, J. Tervonen, J. Närväinen, P. Henttonen, I. Määttänen, and J. Mäntyjärvi, "Selecting feature sets and comparing classification methods for cognitive state estimation," in *2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE)*. IEEE, 2020, pp. 683–690.

[12] T. Smeets, S. Cornelisse, C. W. Quaedflieg, T. Meyer, M. Jelicic, and H. Merckelbach, "Introducing the maastricht acute stress test (mast): a quick and non-invasive approach to elicit robust autonomic and glucocorticoid stress responses," *Psychoneuroendocrinology*, vol. 37, no. 12, pp. 1998–2008, 2012.

[13] R. K. Nath, J. Tervonen, J. Närväinen, K. Pettersson, and J. Mäntyjärvi, "Acute stress data-based fast biometric system using contrastive learning and ultra-short ecg signal segments," in *Adjunct Proceedings of the 2023 ACM International Joint Conference on Pervasive and Ubiquitous Computing the 2023 ACM International Symposium on Wearable Computing (UbiComp/ISWC '23 Adjunct), October 8–12, 2023, Cancun, Quintana Roo, Mexico*, 2023.